SMART DISASTER DETECTION AND ALERTING SYSTEM

A Mini Project Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Electronics and Communication Engineering

by

AADIL MUHAMMED(MBC22EC001)

BEVAL PHILIP MATHEW(MBC22EC018)

RAHUL B(MBC22EC045)

SIBI B JOHN(MBC22EC054)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
MAR BASELIOS CHRISTIAN COLLEGE OF ENGINEERING AND TECHNOLOGY, PEERMADE

April 2025

SMART DISASTER DETECTION AND ALERTING SYSTEM

A Mini Project Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Electronics and Communication Engineering

by

AADIL MUHAMMED(MBC22EC001)

BEVAL PHILIP MATHEW(MBC22EC018)

RAHUL B(MBC22EC045)

SIBI B JOHN(MBC22EC054)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

MAR BASELIOS CHRISTIAN COLLEGE OF ENGINEERING AND TECHNOLOGY, PEERMADE

April 2025

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

MAR BASELIOS CHRISTIAN COLLEGE OF ENGINEERING AND TECHNOLOGY

2024-2025

CERTIFICATE

This is to certify that this report entitled **"SMART DISASTER DETECTION AND ALERTING SYSTEM"** submitted by **AADIL MUHAMMED (MBC22EC 001), BEVAL PHILIP MATHEW (MBC22EC018), RAHUL B (MBC22EC045) SIBI B JOHN (MBC22EC054)** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of B.Tech. in Electronics and Communication Engineering,is a bonafide report of the mini project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Prof.Nimisha George	**Prof.Nimisha George**	**Prof.Anu Mary Mathew**
Mini Project Guide	Mini Project Coordinator	Head of the Dept
Assistant Professor	Assistant Professor	Associate Professor
Dept. of ECE	Dept. of ECE	Dept. of ECE

ABSTRACT

The Smart Disaster Detection and Alerting System is an early warning system that monitors and alerts about the slightest vibrations, moisture content, and tilt that occur in the soil. By the collected data, we can alert the authorities and the corresponding communities about the chances of an earthquake or landslide occurring in those areas. With these kinds of early warning systems, we can effectively take proper safety measures and reduce the impacts.

The main sensors we are using here are the moisture sensor for detecting the water content in the soil, the vibration sensor for monitoring vibrations, and the gyro accelerometer, which checks for tilts in the soil. We are using the ESP32 microcontroller for data processing and buzzers for alerting affected areas. Real-time monitoring and corresponding authority alerting are provided through BLYNK IoT.

The data collected by the sensors are processed by the microcontroller, and if the readings exceed threshold values, alerts are triggered in the affected areas and to the corresponding authorities. All collected data are transmitted in real-time to the authorities for monitoring. Even the slightest variations in readings can be identified and monitored.

ACKNOWLEDGEMENT

We take this opportunity to express our deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere thanks to **Dr. V I George, Principal** and to **Prof. Elias Janson K, Vice Principal**, Mar Baselios Christian College of Engineering and Technology, Peermade, for their valuable support.We would also like to thank **Prof. Anu Mary Mathew, Associate Professor and HOD** of Electronics and Communication Engineering, for providing us with all the necessary facilities and support. We would like to express our sincere gratitude to our project coordinators **Prof. Nimisha George and Prof. Gishamol R** for their support and coordination. We would like to place on record our sincere gratitude to our mini project guide **Prof. Nimisha George, Assistant Professor**, for the guidance and mentorship throughout the course. Finally, we thank our family and friends who contributed to the successful fulfillment of this project work.

AADIL MUHAMMED
BEVAL PHILIP MATHEW
RAHUL B
SIBI B JOHN

CONTENTS

LIST OF FIGURES

ABBREVIATIONS

IDE - Integrated Development Environment
MCU - Microcontroller Unit
SCL - Serial Clock
SDA - Serial Data
DC - Direct Current
IOT - Internet of Things
I2C - Inter-Integrated Circuit

CHAPTER 1
INTRODUCTION

1.1 General Background

Disasters, both natural and man-made, pose significant threats to human life and infrastructure. Events such as earthquakes, fires, gas leaks, and building collapses can cause severe damage if not detected and addressed promptly. Traditional disaster detection methods often rely on manual monitoring or delayed response systems, which can lead to increased casualties and property loss. With advancements in sensor technology, microcontrollers, and IoT (Internet of Things), it is now possible to develop an intelligent system that can continuously monitor environmental conditions and detect potential disasters in real time. The Smart Disaster Detection and Alerting System is designed to provide early warnings through the integration of multiple sensors, wireless communication, and automated alert mechanisms. By employing technologies such as temperature sensors, gas sensors, fire detection modules, vibration sensors, and wireless alerting systems, this project aims to create a reliable and efficient disaster management tool. The system ensures that hazards are detected early, notifications are sent instantly, and necessary actions can be taken to prevent or mitigate disasters.

1.2 Objective

The main objective of this project is to develop an automated and efficient disaster detection system that can monitor and identify potential hazards while providing real-time alerts. The key goals of the system are:

- Continuous real-time monitoring of environmental parameters such as temperature, smoke, gas concentration, and structural vibrations.
- Early detection of potential hazards using integrated sensors and microcontrollers.
- Instant alert notifications through alarms, SMS, mobile applications, and cloud-based messaging systems to enable quick response.
- User-friendly and scalable design suitable for homes, industries, and public infrastructure.
- Integration of IoT and wireless communication for remote access and monitoring.

The project also aims to incorporate self-sustaining power sources, such as solar energy or battery backups, ensuring uninterrupted operation even in cases of power failure.

Furthermore, cloud-based data logging and analytics can provide valuable insights into disaster trends and risk assessment, aiding in future preparedness and safety measures. The integration of machine learning algorithms for anomaly detection can further enhance the system's ability to distinguish between normal fluctuations and actual hazardous conditions. Additionally, the project focuses on developing a cost-effective and energy-efficient solution, making it accessible and scalable for diverse applications, including smart homes, industries, hospitals, and disaster-prone regions.Ultimately, the system strives to enhance public safety, minimize losses, and contribute to smarter disaster management strategies.

CHAPTER 2
LITERATURE SURVEY

Landslides pose a serious threat to human life and infrastructure, often occurring unpredictably due to environmental factors such as heavy rainfall and earthquakes. Traditional landslide monitoring methods face challenges such as high costs, limited spatial coverage, and low real-time data resolution. To address these issues, an IoT-driven microseismic sensing system has been developed to enhance real-time monitoring and early warning capabilities.

This system utilizes geophone sensors to detect microseismic vibrations in landslide-prone regions, providing continuous data collection and analysis. The ESP32-based edge computing nodes process the collected signals, distinguishing landslide-related tremors from ambient noise. Real-world field deployments, such as in Chandmari, Sikkim, India, have demonstrated the system's ability to capture landslide precursors and provide timely alerts.

The integration of cloud-based data analytics, IoT communication, and machine learning algorithms ensures that the system offers high accuracy, cost-effectiveness, and scalability. By implementing this solution, disaster management authorities can improve early warnings, reduce response time, and enhance overall landslide mitigation strategies.[**1**]

Earthquakes are among the most devastating natural disasters, causing widespread damage to infrastructure and human life. Traditional earthquake detection methods rely on seismographs, which, while accurate, are expensive and have limited accessibility. To address this, an IoT-driven earthquake alert system has been proposed, integrating accelerometers, microcontrollers (Arduino Uno), and real-time communication networks to provide a cost-effective, scalable, and rapid response solution.

The system utilizes ADXL335 3-axis accelerometers to detect seismic vibrations and analyze movement patterns. When abnormal vibrations are detected, the system cross-references them with predefined threshold values to distinguish between normal environmental disturbances and actual seismic activity. Upon confirmation, the system activates visual and auditory alerts via buzzers and LEDs and displays warnings on an LCD screen.

One of the key advantages of this system is its real-time data processing and remote accessibility through IoT-based connectivity. It enables instant alerts via SMS, cloud platforms, or mobile applications, ensuring that individuals and emergency services receive timely notifications. Additionally, the system is low-cost, energy-efficient, and easily deployable, making it suitable for implementation in earthquake-prone areas, smart cities, and industrial zones.[**2**]

Landslides are a major natural disaster that causes significant damage to life and property, particularly in hilly and mountainous regions such as the Himalayas and Western Ghats. Traditional landslide monitoring systems, such as radar satellite interferometry, LIDAR, and geophysical sensor networks, are either expensive, less accurate, or lack real-time monitoring capabilities.

To address these limitations, an IoT-based Slope Instability Sensing System has been developed. This system integrates low-cost wireless sensors, including strain gauge-based movement sensors, soil moisture sensors, and pore-water pressure sensors, to monitor changes in soil stability. The sensor nodes communicate wirelessly, transmitting real-time data to a central monitoring system. By continuously analyzing environmental parameters, the system provides early warnings of potential landslides, allowing authorities and residents to take preventive measures.

The real-world deployment of this system in Sikkim, India, has demonstrated its effectiveness in monitoring landslide-prone areas. With its scalability, accuracy, and affordability, this system is a significant advancement in disaster prevention, helping mitigate the risks associated with landslides and protecting vulnerable communities.[**3**]

Natural disasters such as floods, fires, and earthquakes cause significant damage to human life and infrastructure. Traditional disaster management systems often rely on manual detection and delayed responses, making them ineffective for real-time alerts. To address this, an IoT-based Disaster Detection and Alert System has been developed to actively monitor environmental conditions and send real-time alerts to users.

The system integrates multiple sensors, including ultrasonic sensors, float switches, rain sensors, flame sensors, and accelerometers, controlled by a NodeMCU (ESP8266) microcontroller. These sensors continuously monitor water levels, rainfall intensity, fire outbreaks, and seismic activity. The system classifies threats into three alert levels: Green, Orange, and Red, depending on the severity of the detected disaster.

Data is transmitted to ThingSpeak (cloud-based platform), and alerts are sent via the IFTTT mobile application to notify users instantly. The system's real-time monitoring and automated alerts help in minimizing casualties and damage by enabling a faster emergency response. Additionally, this cost-effective and scalable solution can be deployed in residential areas, industries, and public spaces, enhancing overall disaster preparedness and management.[**4**]

Landslides pose a significant threat to infrastructure and human lives, especially in mountainous and hilly regions. Traditional monitoring methods rely on manual inspections or costly technologies that lack real-time capabilities. To address these challenges, a Wireless Sensor Network (WSN)-Based Smart Landslide Monitoring Device, known as SMARTMODE, has been developed to provide real-time landslide detection and early warnings.

SMARTMODE is an energy-efficient, context-adaptive, and robust sensor node that continuously monitors slope movements, soil moisture, and vibrations using a network of low-power sensors. It features automatic data acquisition frequency adjustment based on environmental severity and integrates noise filtering techniques to eliminate false alarms. The device is also equipped with solar-powered energy harvesting and backup mechanisms, ensuring uninterrupted operation even in remote locations.

SMARTMODE has been successfully deployed in landslide-prone areas, demonstrating its ability to provide accurate and reliable data for disaster mitigation. By enabling continuous monitoring, real-time alerts, and efficient power management, this system significantly enhances landslide preparedness and reduces potential damage.[**5**]

Natural disasters such as floods, fires, and earthquakes cause significant damage to human life and infrastructure. Traditional disaster management systems often rely on manual detection and delayed responses, making them ineffective for real-time alerts. To address this, an IoT-based Disaster Detection and Alert System has been developed to actively monitor environmental conditions and send real-time alerts to users.

The system integrates multiple sensors, including ultrasonic sensors, float switches, rain sensors, flame sensors, and accelerometers, controlled by a NodeMCU (ESP8266) microcontroller. These sensors continuously monitor water levels, rainfall intensity, fire outbreaks, and seismic activity. The system classifies threats into three alert levels: Green, Orange, and Red, depending on the severity of the detected disaster.

Data is transmitted to ThingSpeak (cloud-based platform), and alerts are sent via the IFTTT mobile application to notify users instantly. The system's real-time monitoring and automated alerts help in minimizing casualties and damage by enabling a faster emergency response. Additionally, this cost-effective and scalable solution can be deployed in residential areas, industries, and public spaces, enhancing overall disaster preparedness and management.[**6**]

CHAPTER 3
PROPOSED SYSTEM

3.1 Proposed Work

The Smart Disaster Detection and Alerting System is designed to provide automated and real-time monitoring of environmental conditions to detect disasters such as earthquakes and landslides. The system integrates advanced sensors, microcontrollers, and IoT-based communication to enhance disaster preparedness and minimize response time. At the core of the system, an ESP32 microcontroller continuously processes data collected from various sensors, including an MPU6050 Gyro Accelerometer, an SW-420 Vibration Sensor, and a Soil Moisture Sensor. These sensors play a crucial role in detecting seismic activity, ground vibrations, and moisture levels, which are key indicators of potential disasters.

When the system detects abnormal sensor readings that exceed predefined safety thresholds, it automatically triggers an alert to notify relevant authorities and nearby individuals. This is achieved through Blynk IoT notifications, which send real-time updates to mobile devices, and a buzzer alarm, which provides an immediate audible warning. The use of IoT-based remote monitoring allows for continuous observation of disaster-prone areas, enabling timely interventions and preventive measures.

The system is cost-effective, scalable, and energy-efficient, making it suitable for deployment in homes, industries, and public infrastructure. Its ability to automate hazard detection, provide instant alerts, and facilitate quick response actions significantly enhances disaster management efforts. By ensuring continuous monitoring and real-time communication, this system plays a vital role in reducing risks associated with earthquakes and landslides, ultimately improving public safety and disaster preparedness.

3.2 Block Diagram

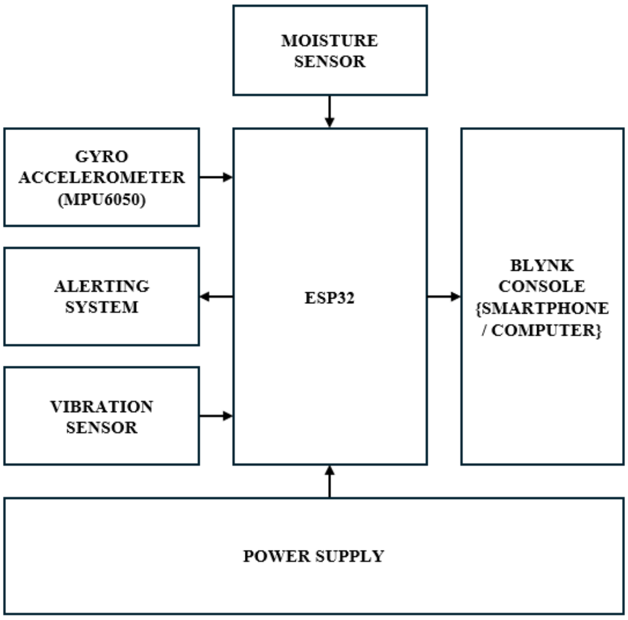


FIG 3.2:Block Diagram

The block diagram represents an IoT-based Landslide and Earthquake Detection, Alerting, and Monitoring System using the ESP32 microcontroller. This system is designed to monitor environmental conditions and detect early warning signs of landslides and earthquakes, helping to minimize risks and enhance disaster preparedness. By integrating various sensors, the system continuously collects real-time data, analyzes potential hazards, and provides alerts when necessary.

The system consists of three primary sensors: a moisture sensor, a gyro accelerometer (MPU6050), and a vibration sensor. The moisture sensor measures the soil's moisture content, which is a crucial factor in landslides, as excessive moisture weakens soil structure. The gyro accelerometer detects changes in tilt and motion, indicating ground shifts that may signal instability. The vibration sensor monitors seismic activity, identifying tremors that could be early signs of an earthquake.

The ESP32 microcontroller processes the collected data and determines whether the conditions indicate a potential landslide or earthquake. If an abnormality is detected, the alerting system is triggered, notifying relevant authorities or individuals in the affected area. Simultaneously, the system transmits real-time data to a Blynk console, allowing users to remotely monitor seismic and environmental conditions through a smartphone or computer.

A dedicated power supply ensures that the system remains operational even in critical conditions, making it a reliable tool for disaster detection. This IoT-based solution is particularly useful in landslide-prone and earthquake-prone regions, where early warnings can help prevent damage and save lives. By providing continuous monitoring and instant alerts, the system plays a vital role in disaster management, risk mitigation, and public safety.

3.3 Components Required

3.3.1 ESP-32 Microcontroller

FIG 3.3:ESP-32

The ESP32 is a powerful and versatile microcontroller that features a dual-core 32-bit Tensilica Xtensa LX6 processor with a clock speed of up to 240 MHz, making it highly efficient for real-time data processing. It comes with integrated Wi-Fi and Bluetooth (BLE 4.2 and Classic), enabling seamless communication in IoT applications. The ESP32 supports multiple GPIO interfaces, including I2C, SPI, UART, ADC, DAC, and PWM, allowing easy integration with various sensors and peripherals. Designed for low power consumption, it includes multiple sleep modes to optimize energy efficiency, making it suitable for battery-powered applications. Additionally, it features capacitive touch sensors, secure boot, and flash encryption for enhanced security. With support for programming environments such as Arduino IDE, MicroPython, and ESP-IDF, the ESP32 is widely used in smart monitoring, automation, and IoT-based projects due to its cost-effectiveness, high-speed processing, and robust wireless connectivity.

3.3.2 Gyro Accelerometer

FIG 3.4:MPU6050

A Gyro Accelerometer is a motion-sensing device that combines a gyroscope and an accelerometer to measure both angular velocity and linear acceleration. The gyroscope detects rotational movements around different axes, while the accelerometer measures changes in speed and direction, including tilt and orientation. By integrating both sensors, the device provides precise motion tracking and stabilization, making it essential for applications that require real-time movement detection. One of the most commonly used modules is the MPU6050, which is a 6-axis sensor consisting of a 3-axis accelerometer and a 3-axis gyroscope. It communicates with microcontrollers through the I2C interface, making it easy to integrate into various projects. Gyro accelerometers are widely used in smartphones, drones, robotics, gaming controllers, and automotive stability systems. Their ability to provide high-precision motion sensing makes them crucial for navigation, gesture recognition, and self-balancing systems.

3.3.3 Vibration Sensor

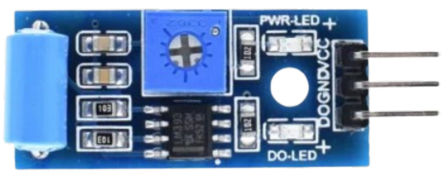


FIG 3.5:Vibration Sensor

The Vibration Sensor is a device used to measure and detect mechanical vibrations, movement, or oscillations in machines, structures, and other applications. These sensors work based on different principles, such as piezoelectric, capacitive, or MEMS technology, to convert vibration into electrical signals. Vibration sensors play a crucial role in predictive maintenance by identifying early signs of mechanical failure in industrial machines, reducing downtime and improving efficiency. They are also used in security systems, automotive safety applications, and structural health monitoring.

3.3.4 Moisture Sensor

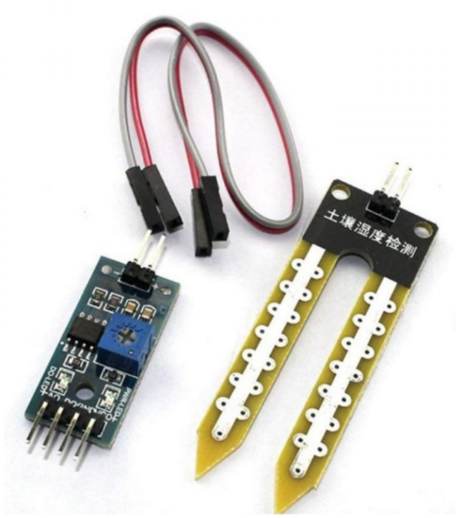


FIG 3.6:Moisture Sensor

The Moisture Sensor is a device used to measure the water content in soil, air, or other materials. These sensors work by detecting changes in electrical resistance, capacitance, or conductivity caused by moisture levels. In agriculture, moisture sensors help optimize irrigation by ensuring crops receive the right amount of water, improving efficiency and crop yield. They are also widely used in industrial applications, environmental monitoring, and smart home automation systems to detect leaks and humidity levels.

CHAPTER 4
HARDWARE DESCRIPTION

4.1 Circuit Diagram

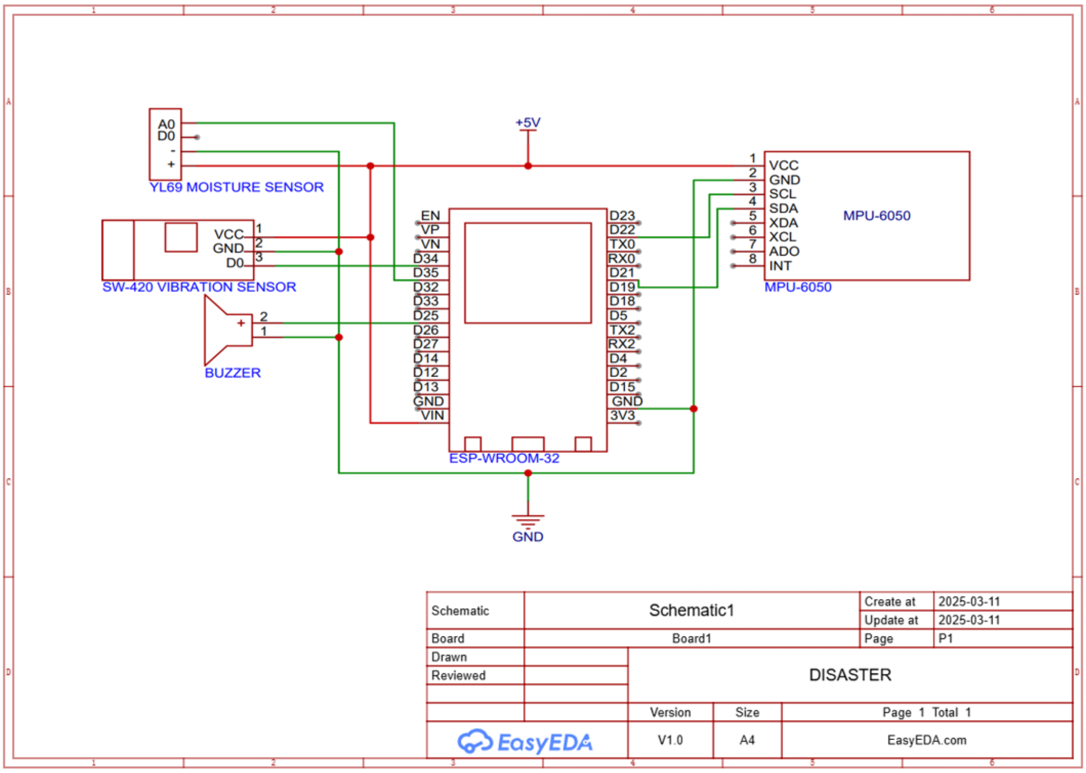


FIG 4.1:Circuit Diagram

The given circuit consists of various interconnected components, with the ESP-32 microcontroller serving as the central unit responsible for processing data and controlling the connected devices. The YL69 Moisture Sensor is connected with its A0 pin linked to an analog input of the ESP32, D0 connected to a digital input, VCC receiving a +5V power supply, and GND linked to the common ground. The SW-420 Vibration Sensor is similarly powered, with VCC connected to +5V, GND to ground, and its D0 pin interfacing with a digital input of the ESP32 for vibration detection. The MPU-6050 Accelerometer and Gyroscope Sensor operate via the I2C communication protocol, with its VCC connected to +5V, GND to ground, SCL linked to an I2C clock pin of the ESP32, and SDA connected to an I2C data pin. Additionally, a buzzer is included in the circuit, where its positive terminal is connected to a digital output pin of the ESP32, and the negative terminal is grounded. The entire system is powered by a +5V power supply, which provides the necessary voltage for the ESP32, sensors, and buzzer, while the GND serves as the common ground for all components.

4.2 EasyEDA Software

EasyEDA is a web-based PCB design and circuit simulation tool that allows users to create, edit, and test electronic circuits online. It provides a user-friendly interface for designing schematics, simulating circuits, and generating PCB layouts. The platform supports a vast library of electronic components and modules, making it easier to design complex circuits efficiently. EasyEDA also integrates with PCB manufacturing services, allowing users to directly order custom PCBs after completing their designs. It supports importing and exporting files in multiple formats, making collaboration and sharing designs more convenient. In our project, we used EasyEDA to design and simulate the circuit before actual implementation. This helped in identifying potential issues, optimizing connections, and ensuring that the circuit worked as expected before hardware assembly.

CHAPTER 5
SOFTWARE DESCRIPTION

5.1 Flow Chart

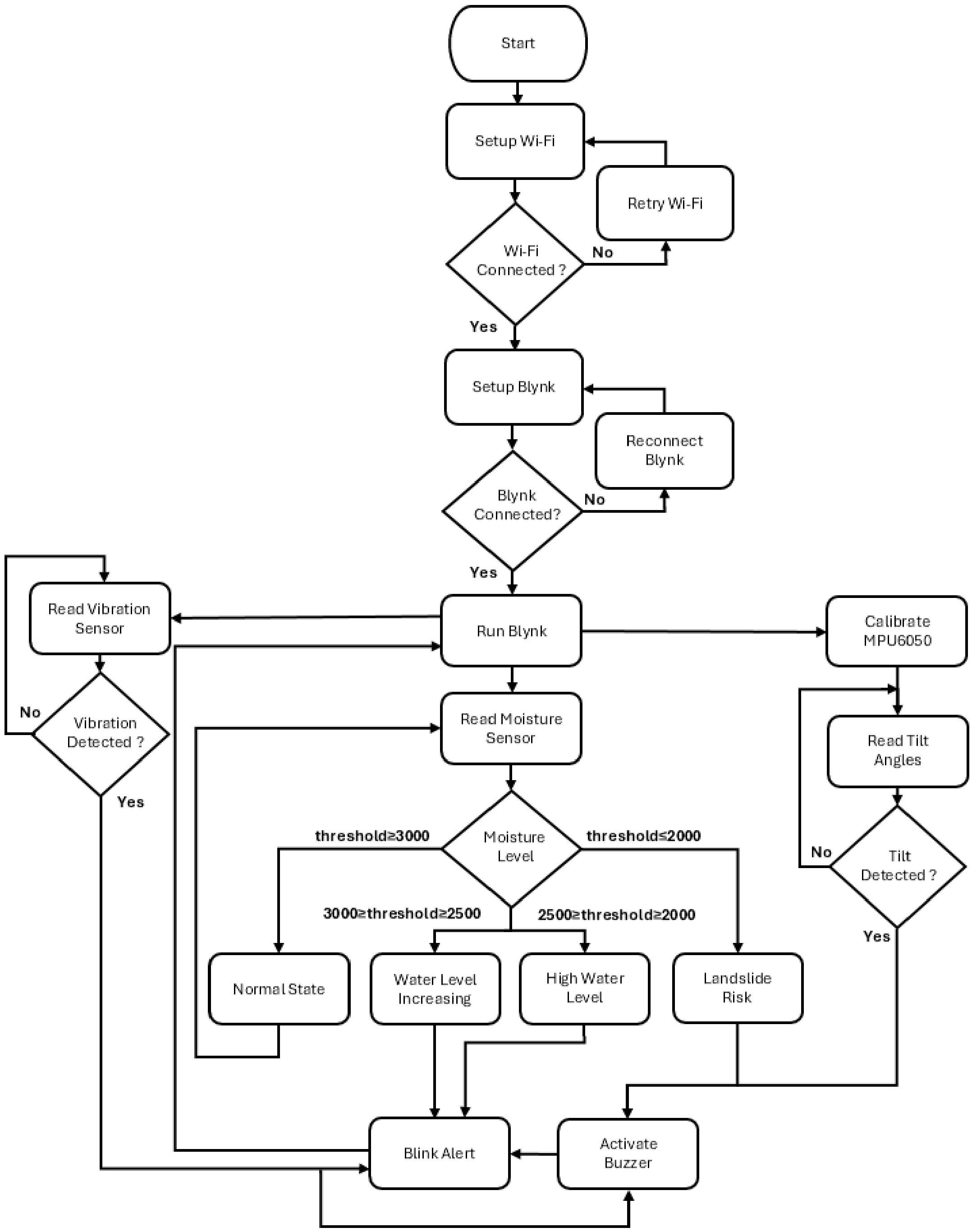


FIG 5.1:Flow Chart

5.2 Arduino IDE

Arduino IDE is an open-source software used for writing, compiling, and uploading code to microcontrollers like Arduino, ESP32, and ESP8266. It has a simple and easy-to-use interface with built-in libraries, making it convenient to work with different sensors and modules. The software supports C and C++ programming and comes with a Serial Monitor for debugging and viewing real-time data. To use Arduino IDE, we need to download it from the official Arduino website and set up the required board configurations, like adding the ESP32 Board Manager. The code mainly consists of two functions: setup(), which runs once to initialize hardware, and loop(), which continuously runs the main logic. In our project, we used Arduino IDE to program the ESP32 for processing data from moisture, vibration, and motion sensors while also controlling the buzzer for alerts. It helped us in debugging and testing the system effectively.

5.3 Blynk IOT

Blynk IoT is a powerful platform that allows users to control and monitor IoT devices remotely using a smartphone or web dashboard. It supports various microcontrollers like Arduino, ESP32, and ESP8266, enabling real-time data visualization and automation. The platform provides an easy-to-use mobile app where users can create custom dashboards with widgets like buttons, graphs, and sliders to interact with connected devices. Blynk works on a cloud-based architecture, allowing devices to communicate over the internet using Wi-Fi, Ethernet, or GSM. It uses the Blynk library, which integrates with Arduino IDE for programming and connecting devices to the Blynk Cloud. In our project, we used Blynk to remotely monitor sensor data and trigger alerts based on specific conditions. The real-time updates and control features of Blynk helped in making our system more efficient and user-friendly.

CHAPTER 6
WORKING

The Smart Disaster Detection and Alerting System is an IoT-based monitoring solution designed to detect environmental hazards like landslides and earthquakes. It utilizes an ESP-WROOM-32 (ESP32) microcontroller to collect real-time data from sensors and provide alerts via a buzzer and the Blynk cloud platform for remote monitoring. The system begins with Wi-Fi initialization, retrying until successful. Once connected, it logs data and sends alerts. If the connection is lost, it attempts to reconnect.

After initialization, the system continuously monitors environmental conditions using three key sensors:

1. YL-69 Moisture Sensor: Detects soil moisture levels, categorized into:

- **Normal (threshold >= 3000):** No immediate threat.
- **Water Increasing (2500-3000):** Rising moisture, potential instability.
- **High Water (2000–2500):** Critical warning; high saturation increases landslide risk.
- **Landslide Risk (threshold <= 2000):** Immediate alert due to excessive moisture.

2. SW-420 Vibration Sensor:
Detects ground vibrations from earthquakes or landslides. If significant movement occurs, the system checks other sensor data to assess disaster likelihood.

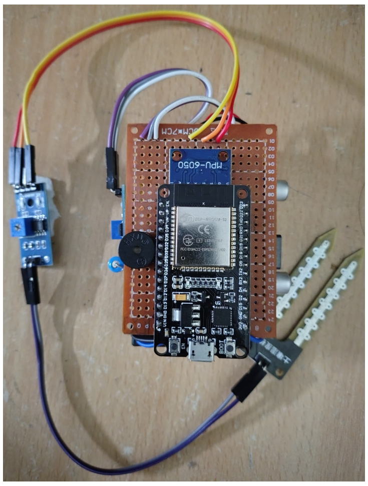
3. MPU-6050 Accelerometer and Gyroscope:
Measures tilt angles to detect land shifts. An abnormal tilt classifies the event as a potential landslide.

If any sensor readings exceed predefined thresholds, the system activates a buzzer to alert individuals and sends real-time notifications via the Blynk platform. This IoT-based solution provides an effective early warning system for disaster-prone areas, enhancing risk mitigation and public safety.

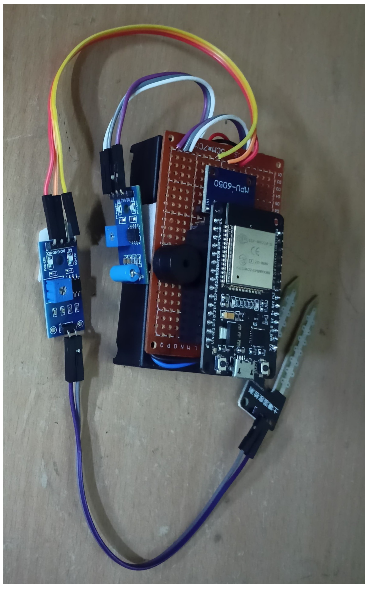
CHAPTER 7
RESULTS

As a result of this project, we successfully developed a Smart Disaster Detection and Alerting System capable of detecting seismic vibrations and moisture levels to predict disasters such as earthquakes and landslides. The system utilizes sensors, including a gyro accelerometer, vibration sensor, and moisture sensor, with data processing handled by an ESP32 microcontroller. The collected data is analyzed against predefined threshold values, triggering alerts when abnormal readings are detected. These alerts are transmitted via the Blynk IoT platform and a buzzer to notify relevant authorities and communities in real time.

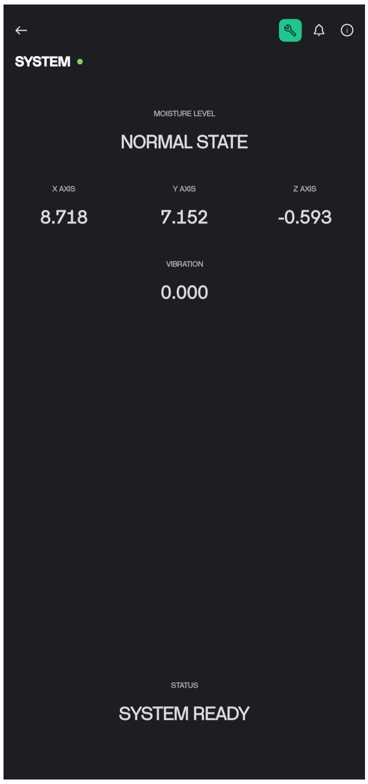
While the system effectively detects potential disasters, there were some limitations. The response time of the alerting mechanism depends on network connectivity, which may cause slight delays. Additionally, environmental noise and sensor calibration challenges slightly affect accuracy. However, the system successfully demonstrated the feasibility of real-time disaster detection on a low-cost and scalable platform, providing valuable insights for future improvements.



(a) *FIG 7.1:Top view*



(b) *FIG 7.2:Side view*



(c) *FIG 7.3:Blynk Console*

The system was able to detect changes in vibration and moisture levels efficiently.

Upon activation, the ESP32 connected to the sensors and monitored environmental conditions. When a threshold breach was identified, the system triggered an alert via the Blynk app and activated the buzzer. Testing showed that the system could recognize sudden vibrations and moisture variations within seconds, making it an effective early warning tool.

Although the project was successful, enhancements such as advanced sensor integration, machine learning-based prediction models, and improved network redundancy can be explored to further optimize the system. The results confirm the potential of IoT-based disaster detection solutions in mitigating risks and ensuring timely interventions.

CHAPTER 8
DISCUSSIONS

8.1 Advantages

- Real-Time Monitoring – Continuous data collection on seismic activity, land moisture, and vibrations.
- Automated Detection – Uses MPU6050, vibration sensor, and moisture sensor for high accuracy.
- Instant Alerts – Sends notifications via Blynk IoT and buzzer, reducing response time.
- Scalability – Deployable across multiple locations for widespread monitoring.
- Low Cost – Affordable components and open-source software reduce expenses.
- Energy-Efficient – Can integrate solar power for sustainable operation.
- IoT Connectivity – Remote monitoring via mobile applications.
- AI and ML Integration Potential – Future versions can enhance disaster prediction.
- Community Safety Enhancement – Reduces life and property loss through quick response.
- Reliable Communication – Can integrate GSM/GPS for improved location tracking.

8.2 Disadvantages

- Environmental Limitations – Sensor accuracy can be affected by extreme weather.
- Power Dependency – Requires a stable power source for continuous operation.
- Network Dependency – Internet issues may delay alerts.
- False Alarms – May trigger unnecessary warnings due to minor disturbances.
- Limited Coverage – Requires multiple sensors to monitor large areas.
- Maintenance Requirements – Periodic calibration and replacement needed.
- Data Processing Limitations – Depends on ESP32 processing capabilities.

8.3 Applications

- Earthquake and Landslide Detection – Early detection of seismic activities.
- Government and Disaster Management – Real-time data for decision-making.

- Smart Cities and Infrastructure Safety – Enhances urban safety.
- Remote and Hilly Areas – Useful for landslide and earthquake-prone regions.
- Agriculture and Water Management – Monitors soil moisture to prevent waterlogging.
- IoT-Based Environmental Monitoring – Useful for environmental research.
- Mining and Construction Sites – Ensures worker safety in excavation zones.
- Military and Defense – Detects underground movements for border security.

8.4 Future Scope

To enhance the system's capabilities, future improvements can focus on integrating AI-based predictive analytics for early disaster forecasting. The use of machine learning can improve detection accuracy by analyzing historical data patterns. Additionally, incorporating satellite-based communication can ensure functionality in remote areas where internet connectivity is limited. Expanding the system to monitor additional environmental factors, such as temperature and air pressure, will make it more comprehensive. Developing a mobile application for real-time alerts and emergency response coordination will further improve usability. With these advancements, the system can evolve into a more efficient and globally deployable disaster management solution.

CHAPTER 9
CONCLUSION

9.1 Conclusion

The Smart Disaster Detection and Alerting System successfully automates disaster monitoring by detecting seismic vibrations, land moisture variations, and tilting angles in real time. By leveraging IoT-based communication through Blynk, the system ensures accurate and timely alerts, minimizing delays in disaster response. The integration of ESP32 with sensors like the MPU6050, SW-420, and soil moisture sensors enhances detection reliability. The system provides continuous monitoring, improving data accuracy and decision-making for authorities and communities. This project demonstrates an efficient, low-cost, and scalable solution for disaster detection, significantly reducing the risk of late warnings.

REFERENCES

[1] **N. Kumar M. and M. V. Ramesh**, "*Accurate IoT Based Slope Instability Sensing System for Landslide Detection*," IEEE Sensors Journal, vol. 22, no. 17, pp. 17151-17161, Sept. 1, 2022.Sikkim, North Eastern Himalayas, India 10.1109/JSEN.2022.3189903.

[2] **N. V. Teja, S. P. Akshay, K. A. Khaleed, C. S. Kumar, G. P. Manoj Kumar, and R. M. Bhavadharini,** "*An Earthquake Alert System Using Internet of Things*," 2024 10th International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 2024, pp. 1621-1626. doi: 10.1109/ICCSP60870.2024.10544248.

[3] **P. K. Indukala, U. G. Gosh, and M. Vinodini Ramesh,** "*IoT-Driven Microseismic Sensing System and Monitoring Platform for Landslide Detection* Chandmari, Sikkim, North-Eastern Himalayan Region, India," IEEE Access, vol. 12, pp. 97787-97805, 2024. doi: 10.1109/ACCESS.2024.3427386.

[4] **D. Jadhav, S. Jadhav, J. Shelke, D. Kakade, and A. Talele,** "*Disaster Detection System and Alert Notification Using IoT*," 2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU), Bhubaneswar, India, 2024, pp. 1-8. doi: 10.1109/IC-CGU58078.
2024.10530828

[5] **A. Joshi, D. P. Kanungo, and R. K. Panigrahi**, "*WSN-Based Smart Landslide Monitoring Device,*" IEEE Transactions on Instrumentation and Measurement, vol. 72, pp. 1-12, 2023, Art. no. 5501912. doi: 10.1109/TIM.
2023.3269746.

[6] **V. Koushik, N. V. S. Sasipreetham, M. Nithya, and M. P. V**, "*Early Detection and Warning System for Earthquakes Using Internet of Things*," 2024 3rd International Conference on Sentiment Analysis and Deep Learning (ICSADL), Bhimdatta, Nepal, 2024, pp. 529-533. doi: 10.1109/ICSADL61749.2024.00092.

APPENDICES 1

Esp-32 Datasheet

ESP32-WROOM-32 is a powerful, generic Wi-Fi + Bluetooth® + Bluetooth LE MCU module that targets a wide variety of applications, ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming and MP3 decoding.

At the core of this module is the ESP32-D0WDQ6 chip*. The chip embedded is designed to be scalable and adaptive. There are two CPU cores that can be individually controlled, and the CPU clock frequency is adjustable from 80 MHz to 240 MHz. The chip also has a low-power coprocessor that can be used instead of the CPU to save power while performing tasks that do not require much computing power, such as monitoring of peripherals. ESP32 integrates a rich set of peripherals, ranging from capacitive touch sensors, SD card interface, Ethernet, high-speed SPI, UART, I2S, and I2C.

Note:
* For details on the part numbers of the ESP32 family of chips, please refer to the document *ESP32 Datasheet*.

The integration of Bluetooth, Bluetooth LE and Wi-Fi ensures that a wide range of applications can be targeted, and that the module is all-around: using Wi-Fi allows a large physical range and direct connection to the Internet through a Wi-Fi router, while using Bluetooth allows the user to conveniently connect to the phone or broadcast low energy beacons for its detection. The sleep current of the ESP32 chip is less than 5 μA, making it suitable for battery powered and wearable electronics applications. The module supports a data rate of up to 150 Mbps, and 20 dBm output power at the antenna to ensure the widest physical range. As such the module does offer industry-leading specifications and the best performance for electronic integration, range, power consumption, and connectivity.

The operating system chosen for ESP32 is freeRTOS with LwIP; TLS 1.2 with hardware acceleration is built in as well. Secure (encrypted) over the air (OTA) upgrade is also supported, so that users can upgrade their products even after their release, at minimum cost and effort.

Table 1 provides the specifications of ESP32-WROOM-32.

Table 1: ESP32-WROOM-32 Specifications

Categories	Items	Specifications
Certification	RF certification	See certificates for ESP32-WROOM-32
	Wi-Fi certification	Wi-Fi Alliance
	Bluetooth certification	BQB
	Green certification	RoHS/REACH
Test	Reliablity	HTOL/HTSL/uHAST/TCT/ESD
Wi-Fi	Protocols	802.11 b/g/n (802.11n up to 150 Mbps)
		A-MPDU and A-MSDU aggregation and 0.4 μs guard interval support
	Center frequency range of operating channel	2412 ~ 2484 MHz
Bluetooth	Protocols	Bluetooth v4.2 BR/EDR and Bluetooth LE specification
	Radio	NZIF receiver with –97 dBm sensitivity
		Class-1, class-2 and class-3 transmitter
		AFH

Categories	Items	Specifications
	Audio	CVSD and SBC
Hardware	Module interfaces	SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, Two-Wire Automotive Interface (TWAI®), compatible with ISO11898-1 (CAN Specification 2.0)
	Integrated crystal	40 MHz crystal
	Integrated SPI flash	4 MB
	Operating voltage/Power supply	3.0 V ~ 3.6 V
	Operating current	Average: 80 mA
	Minimum current delivered by power supply	500 mA
	Recommended operating ambient temperature range	–40 °C ~ +85 °C
	Package size	18 mm × 25.5 mm × 3.10 mm
	Moisture sensitivity level (MSL)	Level 3

Pin Definitions

Pin Layout

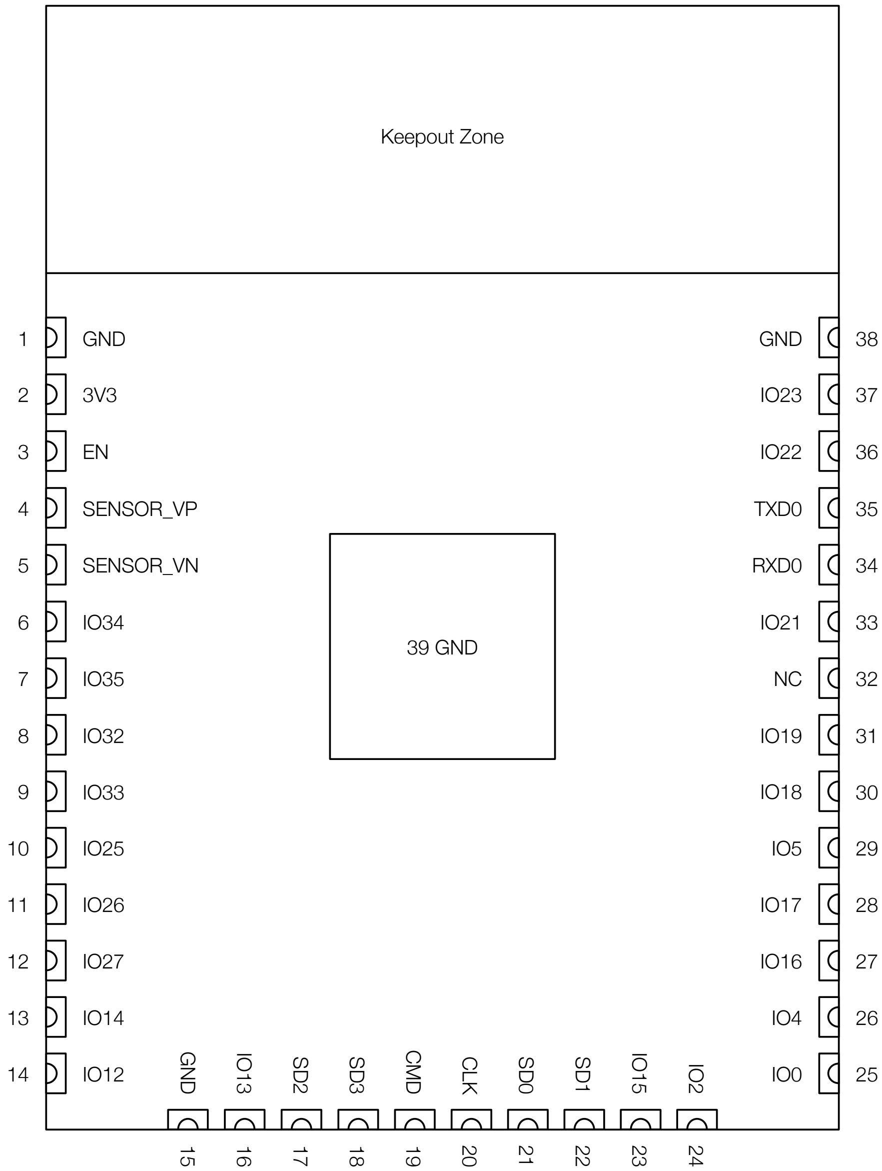


Figure 1: ESP32-WROOM-32 Pin Layout (Top View)

Pin Description

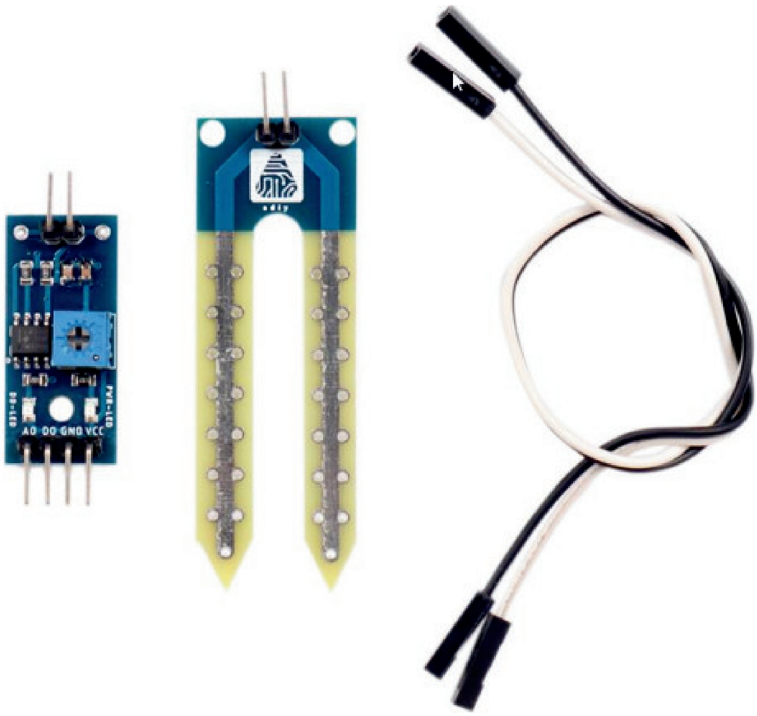
ESP32-WROOM-32 has 38 pins. See pin definitions in Table 2.

Table 2: Pin Definitions

Name	No.	Type	Function
GND	1	P	Ground
3V3	2	P	Power supply
EN	3	I	Module-enable signal. Active high.

Name	No.	Type	Function
SENSOR_VP	4	I	GPIO36, ADC1_CH0, RTC_GPIO0
SENSOR_VN	5	I	GPIO39, ADC1_CH3, RTC_GPIO3
IO34	6	I	GPIO34, ADC1_CH6, RTC_GPIO4
IO35	7	I	GPIO35, ADC1_CH7, RTC_GPIO5
IO32	8	I/O	GPIO32, XTAL_32K_P (32.768 kHz crystal oscillator input), ADC1_CH4, TOUCH9, RTC_GPIO9
IO33	9	I/O	GPIO33, XTAL_32K_N (32.768 kHz crystal oscillator output), ADC1_CH5, TOUCH8, RTC_GPIO8
IO25	10	I/O	GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0
IO26	11	I/O	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
IO27	12	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
IO14	13	I/O	GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2
IO12	14	I/O	GPIO12, ADC2_CH5, TOUCH5, RTC_GPIO15, MTDI, HSPIQ, HS2_DATA2, SD_DATA2, EMAC_TXD3
GND	15	P	Ground
IO13	16	I/O	GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER
SHD/SD2*	17	I/O	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
SWP/SD3*	18	I/O	GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD
SCS/CMD*	19	I/O	GPIO11, SD_CMD, SPICS0, HS1_CMD, U1RTS
SCK/CLK*	20	I/O	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS
SDO/SD0*	21	I/O	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS
SDI/SD1*	22	I/O	GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
IO15	23	I/O	GPIO15, ADC2_CH3, TOUCH3, MTDO, HSPICS0, RTC_GPIO13, HS2_CMD, SD_CMD, EMAC_RXD3
IO2	24	I/O	GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPIWP, HS2_DATA0, SD_DATA0
IO0	25	I/O	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
IO4	26	I/O	GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
IO16	27	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
IO17	28	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
IO5	29	I/O	GPIO5, VSPICS0, HS1_DATA6, EMAC_RX_CLK
IO18	30	I/O	GPIO18, VSPICLK, HS1_DATA7
IO19	31	I/O	GPIO19, VSPIQ, U0CTS, EMAC_TXD0
NC	32	-	-
IO21	33	I/O	GPIO21, VSPIHD, EMAC_TX_EN
RXD0	34	I/O	GPIO3, U0RXD, CLK_OUT2
TXD0	35	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
IO22	36	I/O	GPIO22, VSPIWP, U0RTS, EMAC_TXD1
IO23	37	I/O	GPIO23, VSPID, HS1_STROBE
GND	38	P	Ground

YL-69 Moisture Sensor Datasheet



Description:

This soil moisture sensor module is used to detect the moisture of the soil. It measures the volumetric content of water inside the soil and gives us the moisture level as output. The module has both digital and analog outputs and a potentiometer to adjust the threshold level.

Features and Specifications:

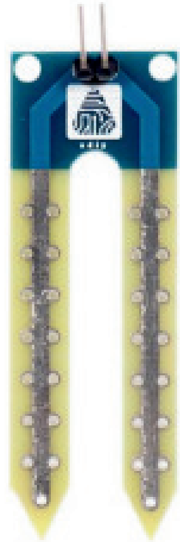
- Operating Voltage: 3.3V to 5V DC
- Operating Current: 15mA
- Output Digital - 0V to 5V, Adjustable trigger level from preset
- Output Analog - 0V to 5V based on infrared radiation from fire flame falling on the sensor
- LEDs indicating output and power
- LM393 based design
- Easy to use with Microcontrollers or even with normal Digital/Analog IC
- Small, cheap and easily available

Hardware Overview:

A typical soil moisture sensor consists of two parts:

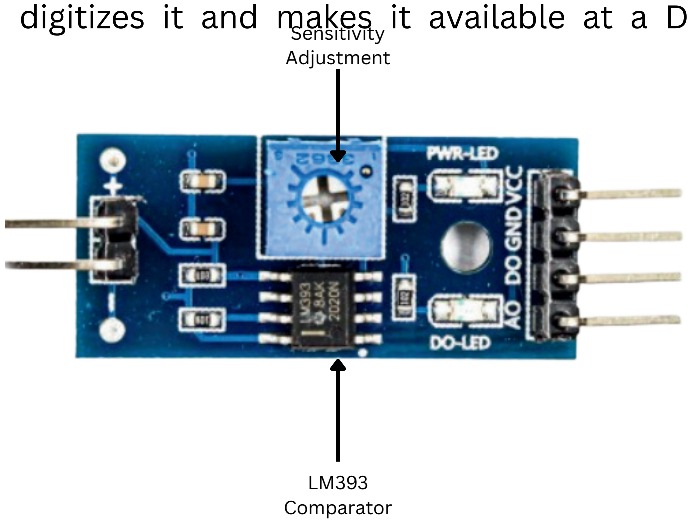
The Probe

The sensor includes a fork-shaped probe with two exposed conductors that is inserted into the soil or wherever the moisture content is to be measured. As previously stated, it acts as a variable resistor, with resistance varying according to soil moisture.



The Module

In addition, the sensor includes an electronic module that connects the probe to the Arduino. The module generates an output voltage based on the resistance of the probe, which is available at an Analog Output (AO) pin. The same signal is fed to an LM393 High Precision Comparator, which digitizes it and makes it available at a Digital Output (DO) pin.



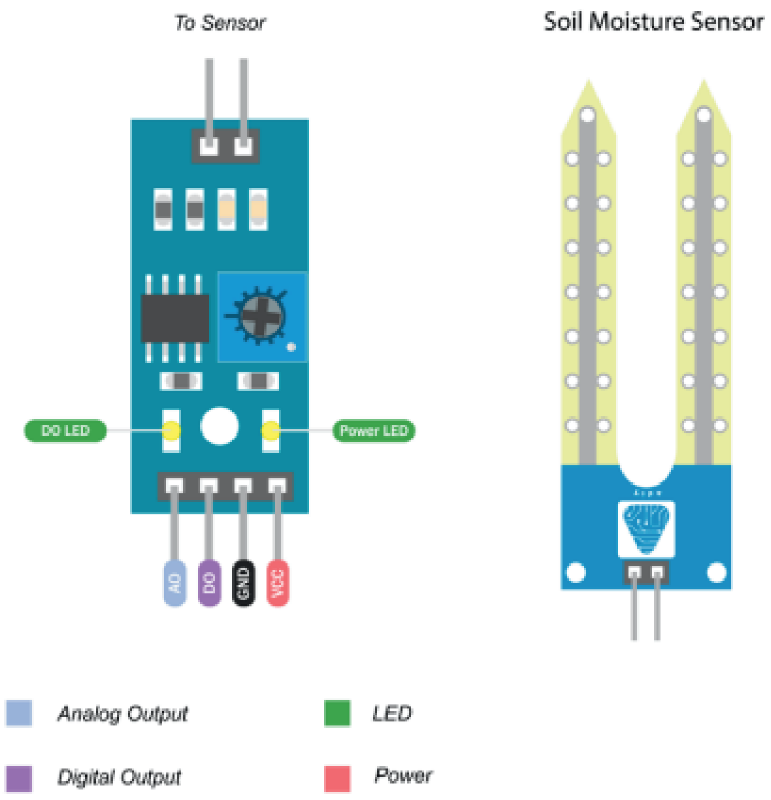
The module includes a potentiometer for adjusting the sensitivity of the digital output (DO).

You can use it to set a threshold, so that when the soil moisture level exceeds the threshold, the module outputs LOW otherwise HIGH.

This setup is very useful for triggering an action when a certain threshold is reached. For example, if the moisture level in the soil exceeds a certain threshold, you can activate a relay to start watering the plant.

The module also includes two LEDs. The Power LED illuminates when the module is turned on, and the Status LED illuminates when the soil moisture level exceeds the threshold value.

Pin Configuration:



VCC: supplies power to the sensor. It is recommended that the sensor be powered from 3.3V to 5V. Please keep in mind that the analog output will vary depending on the voltage supplied to the sensor.

GND: is the ground pin.

AO: Analog Output generates analog output voltage proportional to the soil moisture level, so a higher level results in a higher voltage and a lower level results in a lower voltage.

DO: Digital Output indicates whether the soil moisture level is within the limit. DO becomes LOW when the moisture level exceeds the threshold value (as set by the potentiometer), and HIGH otherwise.

Application:

- Gardening
- Irrigation System
- Used in Controlled Environment

MPU6050 Accelerometer and Gyroscope Datasheet

The MPU6050 module is a The MPU6050 is a Micro Electro-Mechanical Systems (MEMS) which consists of a 3-axis Accelerometer and 3-axis Gyroscope inside it. This helps us to measure acceleration, velocity, orientation, displacement and many other motion related parameter of a system or object. This module also has a (DMP) Digital Motion Processor inside it which is powerful enough to perform complex calculation and thus free up the work for Microcontroller.

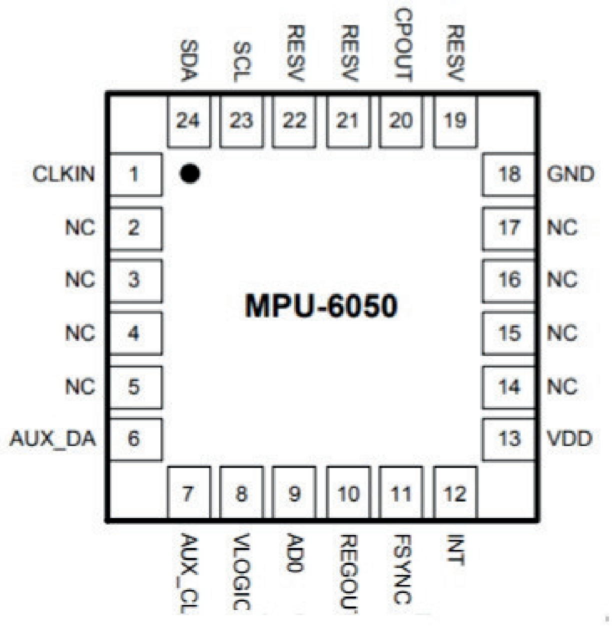
The module also have two auxiliary pins which can be used to interface external IIC modules like an magnetometer, however it is optional. Since the IIC address of the module is configurable more than one MPU6050 sensor can be interfaced to a Microcontroller using the AD0 pin. This module also has well documented and revised libraries available hence it's very easy to use with famous platforms like Arduino. So if you are looking for a sensor to control motion for your RC Car, Drone, Self balancing Robot, Humanoid, Biped or something like that then this sensor might be the right choice for you.

Micro Electro-Mechanical Systems (MEMS) which consists of a 3-axis Accelerometer and 3-axis Gyroscope inside it. This helps us to measure acceleration, velocity, orientation, displacement and many other motion related parameter of a system or object.

The hardware of the module is very simple, it actually comprises of the MPU6050 as the main components as shown above. Since the module works on 3.3V, a voltage regulator is also used. The IIC lines are pulled high using a 4.7k resistor and the interrupt pin is pulled down using another 4.7k resistor.

The MPU6050 module allows us to read data from it through the IIC bus. Any change in motion will be reflected on the mechanical system which will in turn vary the voltage. Then the IC has a 16-bit ADC which it uses to accurately read these changes in voltage and stores it in the FIFO buffer and makes the INT (interrupt) pin to go high. This means that the data is ready to be read, so we use a MCU to read the data from this FIFO buffer through IIC communication. As easy as it might sound, you may face some problem while actually trying to make sense of the data. However there are lots of platforms like Arduino using which you can start using this module in no time by utilizing the readily available libraries explained below.

Pin Diagram



Pin-1- CLKIN- is the optional external reference clock input. This pin is connected to ground when not in use.
Pin-2, Pin-3, Pin-4, Pin-5 are NC pins. These Pins are not connected internally.
Pin-6, AUX_DA, is the I2C master serial data pin. This pin is used for connecting external sensors.
Pin-7, AUX_CL, is the I2C master serial clock. This pin is used for connecting external sensors.
Pin-8, VLOGIC, is the digital I/O supply voltage pin.
Pin-9, AD0, is the I2C slave address LSB pin.
Pin-10, REGOUT, is the regulator filter capacitor connection.
Pin-11, FSYNC, is the frame synchronization digital input. This pin is connected to ground when not used.
Pin-12, INT, is the interrupt digital output pin.
Pin-13, VDD, is the power supply voltage pin.
Pin-14, Pin-15, Pin-16, Pin-17 are NC pin. These pins are not connected internally.
Pin-18, GND, is the power supply ground.
Pin-19 and Pin-21 are the RESV pins. These pins are reserved.
Pin-20, CPOUT, is the charge pump capacitor connection.
Pin-22, is RESV, the reserved pin.
Pin-23, SCL, is the I2C serial clock.
Pin-24, SDA, is the I2C serial data pin

Specifications of MPU6050

MPU6050 is the worlds first integrated 6- axis motion tracking device. Some of the specifications of this module are given below-

- MPU6050 has a 3-axis gyroscope, 3- axis Accelerometer and a Digital motion processor integrated on a single chip.
- It works on the power supply of 3V-5V.
- MPU6050 uses the I2C protocol for communication and transfer of data.
- This module has a built-in 16-bit ADC which provides great accuracy.
- MPU6050 can be interfaced with other IIC devices such as Magnetometers.
- MPU6050 also has an in-built temperature sensor.
- I2C sensor bus helps it to collect data directly from external 3-axis compass, which it uses to provide a complete 9-axis MotionFusion output.
- For manufacturers, MPU6050 eliminates the need for selection, qualification and system-level integration of discrete devices.
- Using its I2C port, non-inertial sensors such as pressure sensor can be interfaced.
- MPU6050 consists of three 16-bits ADC's for digitizing the gyroscope Outputs and three 16-bits ADC's for digitizing the accelerometer outputs.
- A user-programmable gyroscope range and a user-programmable accelerometer range is present for precision tracking of both fast and slow motions.
- An on-chip 1024 byte FIFO buffer is present which helps in lowering the power consumption of the module.
- The need for frequent pooling of sensor output is minimized with the help of on-chip DMP.
- MPU6050 also has an on-chip oscillator with ±1% variation.
- MPU6050 has low-pass filters for gyroscope, accelerometer and temperature sensor.
- VLOGIC reference pin is used to set the logic levels of the I2C interface.
- The user-programmable range of gyroscope present on MPU6050 is ±250, ±500, ±1000 and ±2000°/sec.
- Image, video, and GPS synchronization are supported by the external sync pin of the gyroscope.
- This gyroscope has improved low-frequency noise performance.
- Gyroscope needs 3.6mA of current for operating.
- Low pass filter of the gyroscope is digitally programmable.
- Accelerometer present on MPU6050 operates on 500μA of current.
- The programmable full-scale range of this accelerometer is ±2g, ±4g, ±8g, and 16g.

- The accelerometer can also detect orientation, tap detection.
- User programmable interrupts are present for accelerometer.
- Between accelerometer and gyroscope axes there is a minimal cross-axis sensitivity.
- To communicate with all registers 400kHz fast mode I2C is used.
- DMP present on MPU6050 supports 3D motion processing and gesture recognition algorithms.
- Burst reading is provided for system processor. After reading data from FIFO the system processor enters into low power sleep mode while MPU collects more data.
- Features such as gesture recognition, panning, zooming, scrolling, tap detection, and shack detection are supported by the programmable interrupts.
- MPU6050 also has an optional external clock input of 32.768kHz or 19.2Mhz

Applications of MPU6050

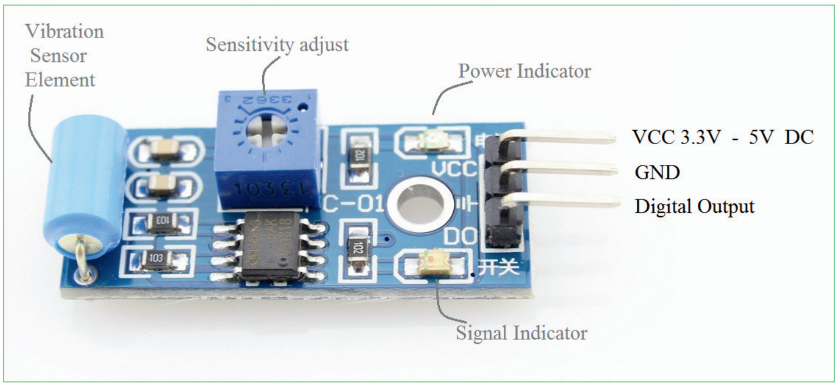
- This module is used in Blurfree technology for video or still image stabilization.
- For recognizing in-air gestures this module is used.
- In the security and authentication systems, MPU6050 is used for gesture recognition.
- For “no-touch” UI application control and navigation MPU6050 is used.
- In motion command technology for gesture short-cuts, this module is used.
- This module has also found application in motion enabled gaming and application frameworks.
- In InstantGesture -IG, MPU6050 is used for gesture recognition.
- Due to its small size, this module is used in handsets and portable gaming equipment.
- Motion-based game controllers also have this module.
- 3D remote controllers, 3D mice also use this module.
- Wearables used for health, fitness and sports also containMPU6050.
- This module can also be found in many toys.
- For IMU measurements MPU6050 is used.
- In drones and quadcopters, MPU6050 is used for position control.
- This module has also found application in self-balancing robots.
- MPU6050 is highly preferred for robotic arm control.
- Humanoid robots also use this module for tilt, rotation, orientation detection.
- In smartphones, this module is used for applications such as augmented reality, gaming, gesture command control, panoramic photo capture, and viewing.
- This module has also being applied for location-based services.

SW-420 Vibration Sensor Datasheet

SW-420 vibration sensor is a module that can detect vibrations or shocks on a surface. It can be used for various purposes, such as detecting door knocks, machine malfunctions, car collisions, or alarm systems. It operates from 3.3 V to 5 V. The module has three peripherals, two LEDs, one for the power status and the other for the sensor output. In addition, there is a potentiometer that can be further used to control the threshold point of the vibration.

SW-420 vibration sensor module consists of a SW-420 vibration switch and an LM393 voltage comparator. A SW-420 vibration switch is a device that has a spring and a rod inside a tube. When the switch is exposed to a vibration, the spring touches the rod and closes the circuit. The vibration sensor in the module detects these oscillations and converts them into electrical signals. The LM393 comparator chip then compares these signals with a reference voltage set by the potentiometer. If the amplitude of the signal exceeds this reference voltage, the output of the comparator goes high (1), otherwise it goes low (0).

Specifications:



This sensor module produce logic states depends on vibration and external force applied on it. When there is no vibration this module gives logic LOW output. When it feels vibration then output of this module goes to logic HIGH. The working bias of this circuit is between 3.3V to 5V DC.

ITEM	VALUE
Operating voltage	3.3V / 5V
Interface	Digital
Size	L: 40mm W: 20mm H: 10mm
Weight	4.3g

VCC: Connects to the power supply (3.3 V to 5 V).
GND: Connects to the ground of the power supply.
DO: Digital output pin that goes high when vibration is detected (after the threshold set by the potentiometer is exceeded).

Electrical Characteristics of SW-420:

- **Voltage <12V**
- **Current <20mA**
- **Closed Resistance <200Ω**
- **Open Resistance >10MΩ**
- **Temperature <80°**

The SW-420 vibration sensor module is able to sense vibration/impact at any level. It's a pretty cheap sensor module that demands little external electronics to get to work. Since it provides a simple HIGH/LOW output (a positive pulse output upon vibration) it can be easily linked with some other electronics circuits. At the same time, you can define the trigger threshold using the onboard trimpot (the trimpot setting may or may not be essential for your intended application, though).
As stated before, if everything goes well, the output of the module is LOW when the module is stable. When a vibration or movement occurs, the module will briefly give a HIGH output

APPENDICES 2

Code for the integration of Esp 32 with the sensors

```
#define BLYNK_TEMPLATE_ID ""
#define BLYNK_TEMPLATE_NAME ""
#define BLYNK_AUTH_TOKEN ""
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <MPU6050_tockn.h>
#include <Wire.h>
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "wifi ssid";
char pass[] = "wifi password";
#define VS_PIN 35   // Vibration Sensor (Analog)
#define MOISTURE_PIN 32 // Moisture Sensor
#define BUZZER_PIN 25 // Buzzer
MPU6050 mpu6050(Wire);
float initialAngleX, initialAngleY, initialAngleZ;
long mpuTimer = 0;
BlynkTimer timer;
bool waterLevelAlertSent = false;
bool landslideRiskAlertSent = false;
bool highWaterAlertSent = false;
bool tiltAlertSent = false;
bool vibrationAlertSent = false;
bool buzzerActive = false;
const float alpha = 0.05;
const int NOISE_THRESHOLD = 10;
const int VIBRATION_THRESHOLD = 100;
float smoothedVibration = 0;
void activateBuzzer() {
if (!buzzerActive) {
digitalWrite(BUZZER_PIN, HIGH);
buzzerActive = true;
}
}
void deactivateBuzzer() {
if (buzzerActive) {
digitalWrite(BUZZER_PIN, LOW);
buzzerActive = false;
}
}
void sendVibrationData() {
int rawVibration = analogRead(VS_PIN);
if (abs(smoothedVibration - rawVibration) > NOISE_THRESHOLD) {
smoothedVibration = (alpha * rawVibration) + ((1 - alpha) *
    smoothedVibration);
}
Blynk.virtualWrite(V6, smoothedVibration);
if (smoothedVibration > VIBRATION_THRESHOLD) {
if (!vibrationAlertSent) {
Blynk.logEvent("vibration_alert", "  High Vibration Detected!");
vibrationAlertSent = true;
}
```

```
activateBuzzer();
} else {
vibrationAlertSent = false;
}
}
void sendMoistureData() {
int sensor_data = analogRead(MOISTURE_PIN);
Blynk.virtualWrite(V1, sensor_data);
if (sensor_data >= 3000) {
Blynk.virtualWrite(V2, "NORMAL STATE");
waterLevelAlertSent = landslideRiskAlertSent = highWaterAlertSent = false;
} else if (sensor_data < 3000 && sensor_data >= 2500) {
if (!waterLevelAlertSent) {
Blynk.virtualWrite(V2, "  WATER LEVEL INCREASING!");
Blynk.logEvent("water_level_increasing", "  Water Level Increasing!");
waterLevelAlertSent = true;
}
} else if (sensor_data < 2500 && sensor_data >= 2000) {
if (!landslideRiskAlertSent) {
Blynk.virtualWrite(V2, "  HIGH WATER LEVEL!");
Blynk.logEvent("high_water_level", "  Immediate Action Required!");
landslideRiskAlertSent = true;
}
} else if (sensor_data < 2000) {
if (!highWaterAlertSent) {
Blynk.virtualWrite(V2, "  LANDSLIDE RISK!");
Blynk.logEvent("landslide_risk", "  Possible Landslide Risk!");
highWaterAlertSent = true;
}
activateBuzzer();
}
}
void sendTiltData() {
mpu6050.update();
if (millis() - mpuTimer > 50) {
float relativeAngleX = mpu6050.getAngleX() - initialAngleX;
float relativeAngleY = mpu6050.getAngleY() - initialAngleY;
float relativeAngleZ = mpu6050.getAngleZ() - initialAngleZ;
Blynk.virtualWrite(V3, relativeAngleX);
Blynk.virtualWrite(V4, relativeAngleY);
Blynk.virtualWrite(V5, relativeAngleZ);
if ((abs(relativeAngleX) > 30 || abs(relativeAngleY) > 30 || abs(
    relativeAngleZ) > 30)) {
if (!tiltAlertSent) {
Blynk.logEvent("tilt_event", "  Tilt Detected!");
tiltAlertSent = true;
}
activateBuzzer();
} else {
tiltAlertSent = false;
}
mpuTimer = millis();
}
}
void checkToDeactivateBuzzer() {
if (!vibrationAlertSent && !highWaterAlertSent && !landslideRiskAlertSent
    && !tiltAlertSent) {
deactivateBuzzer();
```

```
}
}
void setup() {
pinMode(VS_PIN, INPUT);
pinMode(MOISTURE_PIN, INPUT);
pinMode(BUZZER_PIN, OUTPUT);
digitalWrite(BUZZER_PIN, LOW);
WiFi.begin(ssid, pass);
int retry = 0;
while (WiFi.status() != WL_CONNECTED && retry < 20) {
delay(500);
retry++;
}
Blynk.begin(auth, ssid, pass);
Blynk.virtualWrite(V0, "CALIBRATING SENSORS...");
Wire.begin(21, 22);
mpu6050.begin();
mpu6050.calcGyroOffsets(true);
delay(1000);
mpu6050.update();
initialAngleX = mpu6050.getAngleX();
initialAngleY = mpu6050.getAngleY();
initialAngleZ = mpu6050.getAngleZ();
Blynk.virtualWrite(V0, "SYSTEM READY");
timer.setInterval(50L, sendMoistureData);
timer.setInterval(50L, sendTiltData);
timer.setInterval(50L, sendVibrationData);
}
void loop() {
if (!Blynk.connected()) {
Blynk.connect();
}
Blynk.run();
timer.run();
checkToDeactivateBuzzer();
}
```